Arizona Education Data Standard

SIS Vendor Conference and Engagement Kickoff

January 27th 2014

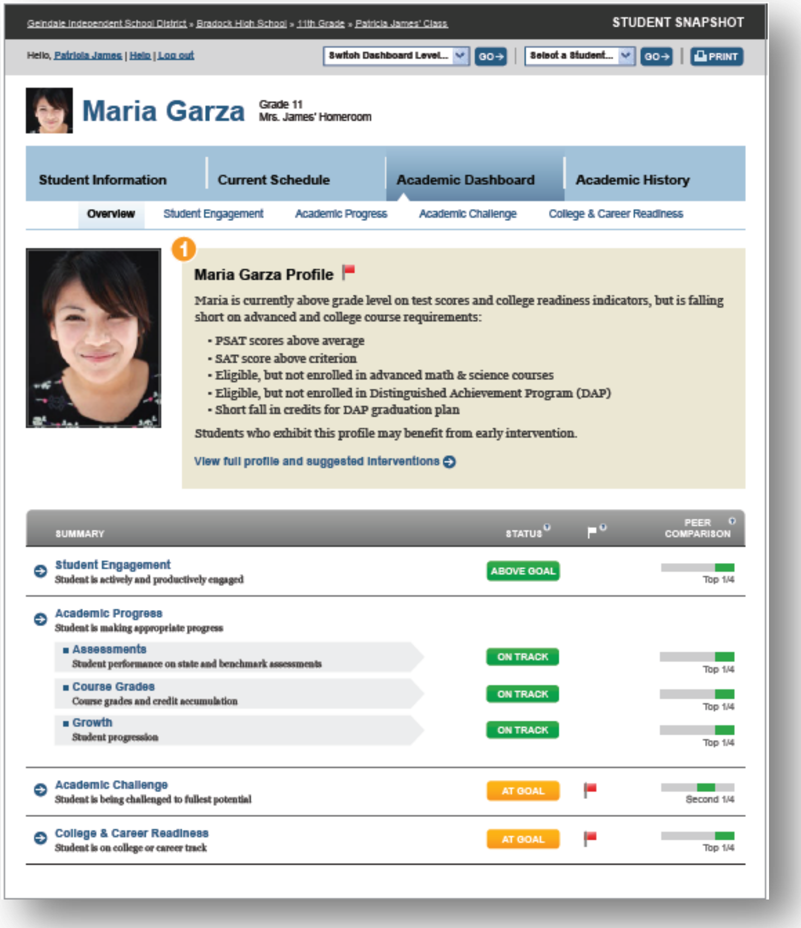
Arizona
Department of Education

Agenda

1. Welcome and Introductions
2. Ed-Fi Alliance Overview
3. Scope and Timeline
4. Focus Areas

Welcome & Introductions

The Power of the Ed-Fi® Solution: Connecting the Data from Start to Finish



Geindale Independent School District » Bradock High School » 11th Grade » Patricia James' Class

Hello, Patricia James | Help | Log out

Maria Garza Grade 11 Mrs. James' Homeroom

Maria Garza Profile

Maria is currently above grade level on test scores and college readiness indicators, but is falling short on advanced and college course requirements:

- PSAT scores above average
- SAT score above criterion
- Eligible, but not enrolled in advanced math & science courses
- Eligible, but not enrolled in Distinguished Achievement Program (DAP)
- Short fall in credits for DAP graduation plan

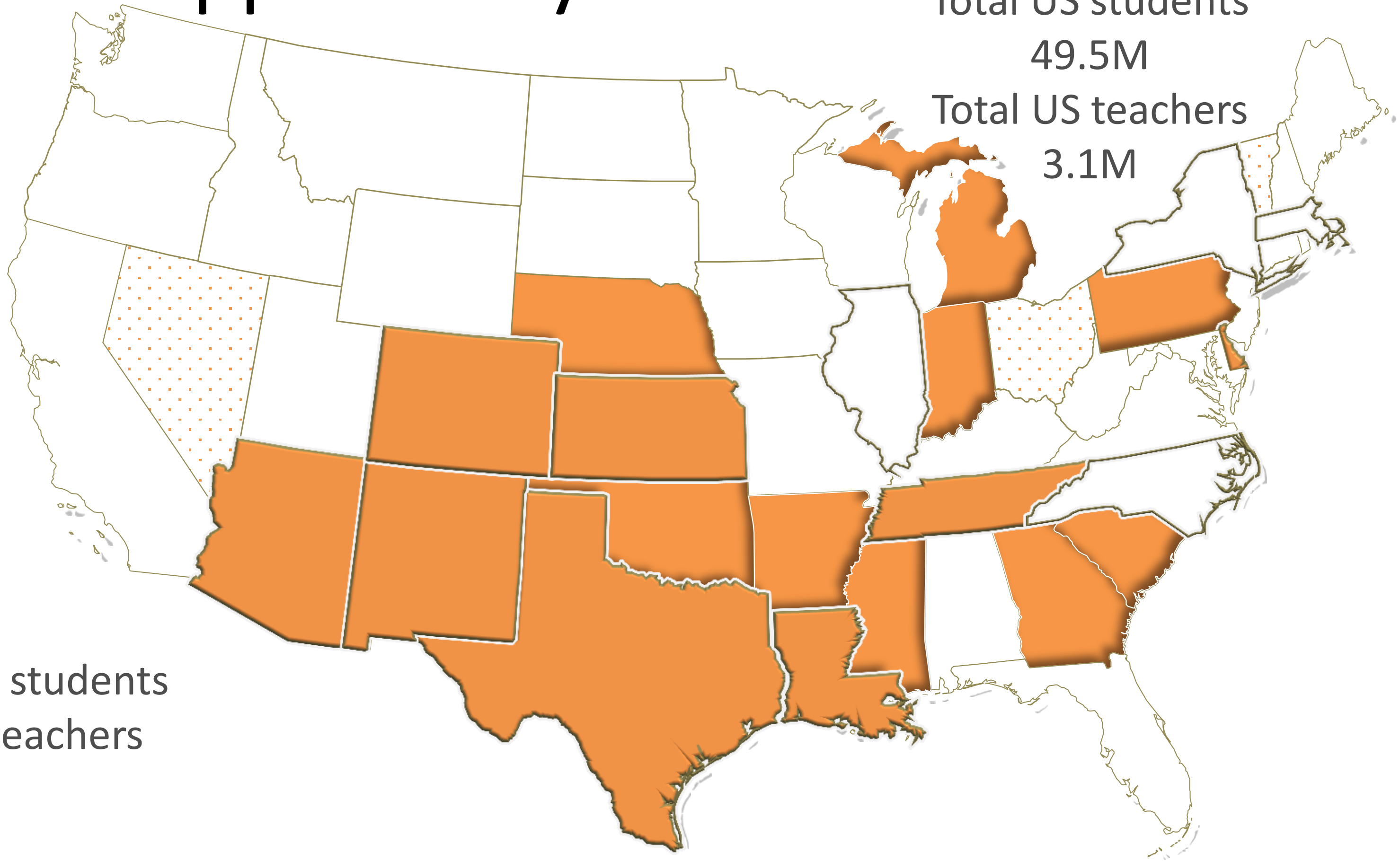
Students who exhibit this profile may benefit from early intervention.

View full profile and suggested interventions

SUMMARY	STATUS	🚩	PEER COMPARISON
Student Engagement Student is actively and productively engaged	ABOVE GOAL		Top 1/4
Academic Progress Student is making appropriate progress			
■ Assessments Student performance on state and benchmark assessments	ON TRACK		Top 1/4
■ Course Grades Course grades and credit accumulation	ON TRACK		Top 1/4
■ Growth Student progression	ON TRACK		Top 1/4
Academic Challenge Student is being challenged to fullest potential	AT GOAL	🚩	Second 1/4
College & Career Readiness Student is on college or career track	AT GOAL	🚩	Top 1/4

- Allows easy, cost-effective implementation of data standards consistent with CEDS
- Provides full-look views into data to support better decisions
- Supports additional use cases to make data more exchangeable
- Enables field-tested community contributions to be broadly, quickly distributed

Growing Ed-Fi Alliance footprint creates significant market opportunity



Total US students
49.5M
Total US teachers
3.1M

🟧 Licensed

▨ In licensing discussions, not included in numbers

18.2M (37%) students
1.2M (38%) teachers

License footprint as of Dec 2013

Market momentum provides significant early adopter advantage

Current and potential customers need solution to interoperability challenges

Participation includes open invitation to shape Alliance certification program

Arizona Education Data Standards (AzEDS)

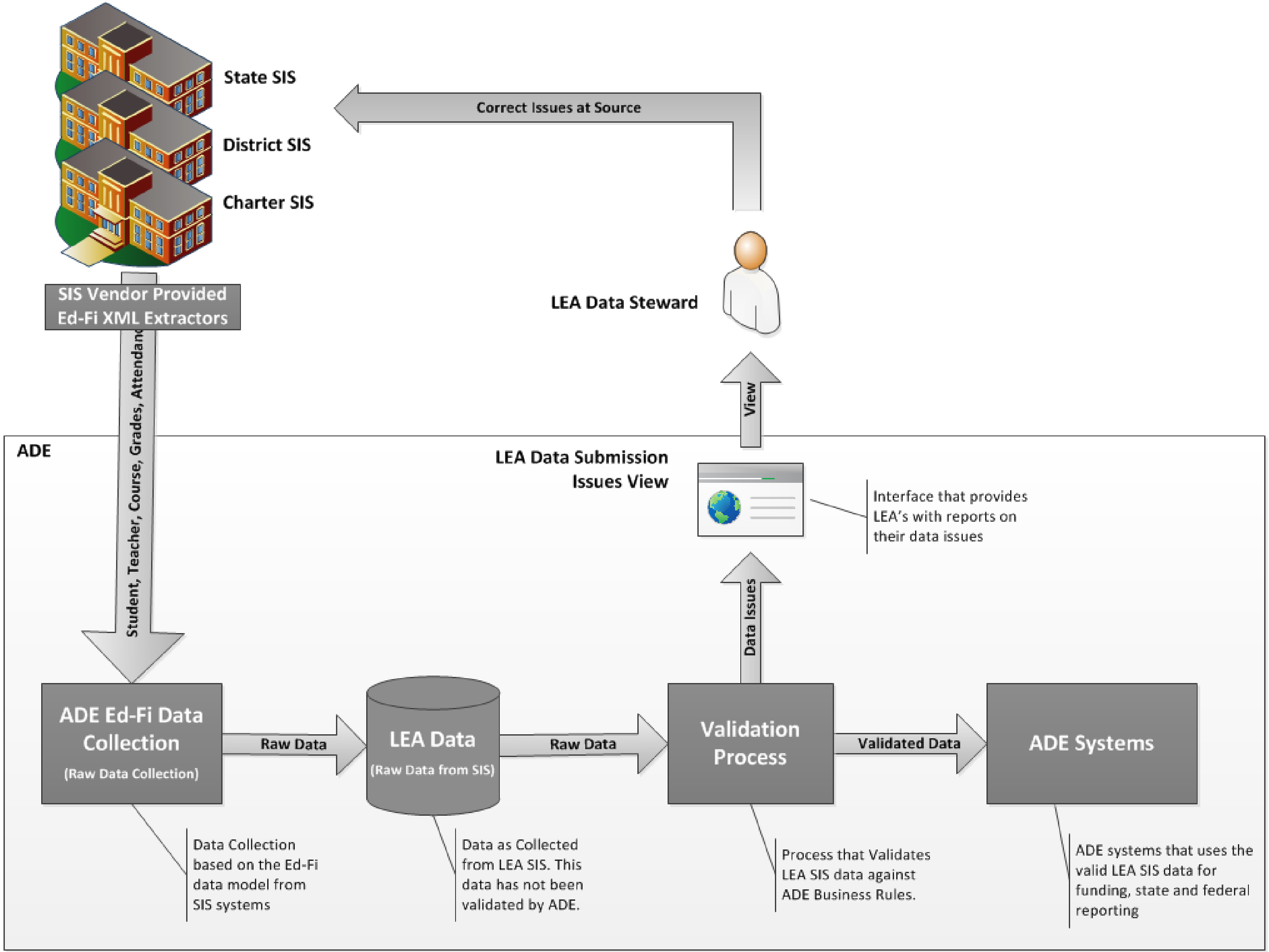
- Set of standards aimed at improving data accuracy & modernizing the state reporting process based on **Ed-Fi® data standard**.
- Allows your district to upload required student-level data to ADE in a reliable, consistent format
  - This single change will eliminate the lot of SAIS data errors every year
- Reduces time spent on ADE data requests & enables near real-time student data to populate the new Arizona Dashboards (AZ Dash)

What we have done so far

- Mapped student-related data elements to Ed-Fi standards and also arrived at extended Ed-Fi mapping for Arizona-specific requirements
  - Consists of student-related data from all program area systems that utilize, gather and report back student-related information
- Analyzed all the processing rules for student-related data (including but is not limited to business and workflow rules)
- Created the high-level design for Web Services, Infrastructure and ETL framework

Arizona Department of Education

High Level Architecture



State SIS

District SIS

Charter SIS

SIS Vendor Provided Ed-Fi XML Extractors

Student, Teacher, Course, Grades, Attendanc

Correct Issues at Source

LEA Data Steward

View

ADE

LEA Data Submission Issues View

Interface that provides LEA's with reports on their data issues

Data Issues

ADE Ed-Fi Data Collection (Raw Data Collection)

Raw Data

LEA Data (Raw Data from SIS)

Raw Data

Validation Process

Validated Data

ADE Systems

Data Collection based on the Ed-Fi data model from SIS systems

Data as Collected from LEA SIS. This data has not been validated by ADE.

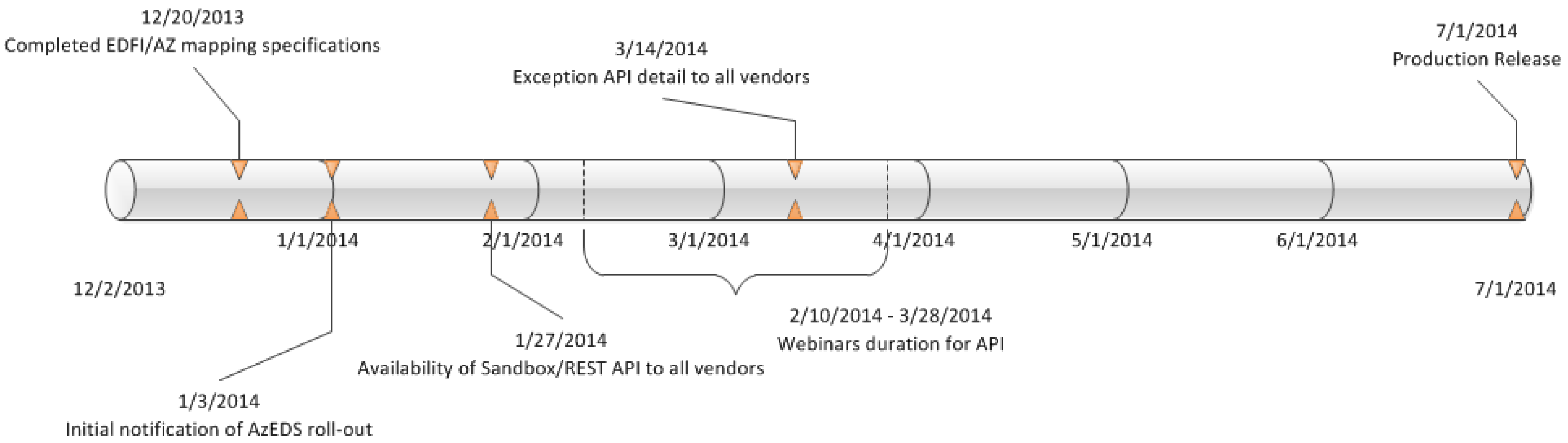
Process that Validates LEA SIS data against ADE Business Rules.

ADE systems that uses the valid LEA SIS data for funding, state and federal reporting

Now vs. Future of data submissions

NOW	FUTURE
SAIS Integrity runs 2x/week	Daily processing of Accountability Rules
250(+) data requests	6 – 12 data requests
Excel formats	REST full Web services
Multiple ODS	One source of truth ODS
ADE business unit specific data elements	Shared data elements across ADE business units

Release 1 – Scope and Timeline


12/20/2013
Completed EDFI/AZ mapping specifications
3/14/2014
Exception API detail to all vendors
7/1/2014
Production Release
12/2/2013
1/1/2014
2/1/2014
3/1/2014
4/1/2014
5/1/2014
6/1/2014
7/1/2014
2/10/2014 - 3/28/2014
Webinars duration for API
1/27/2014
Availability of Sandbox/REST API to all vendors
1/3/2014
Initial notification of AzEDS roll-out

- Release 1 implementation scope:
  - Focusing on Student Enrollment, Attendance, Demographics, Student Teacher Course Connection transactions
  - Ed-Fi REST API – web services to receive student enrollment, attendance, student course connection and other student information from the LEA SIS.
  - LEAData – database to store the data received from the LEA SIS.
  - State Accountability Rules processing service to process submitted student data for any accountability rule violations.
  - Exception Web Service – enables the LEA SIS to pull state accountability rule exceptions on the student data submitted.

Focus Areas

- Collaboration with Double Line Partners (DLP)
  - Developers of Ed-Fi® data standards and the Ed-Fi dashboards
  - Technical support for maintaining the Ed-Fi standards for the Ed-Fi Alliance
- REST API
- Sandbox
- Webinars

REST API

- Documentation of REST API
  - REST API mapping
- Updates to REST API

- Recommended reading on ed-fi.org

Technical Concepts **v1.2**
Outlines the design concepts, principles, patterns and conventions used for the Ed-Fi unifying data model and its associated XML core schema.

Unifying Data Model **v1.2**
Illustrates the Ed-Fi reference model in Unified Modeling Language (UML) class diagrams.

REST API Design Guidelines **v1.2**
Defines a set of design guidelines that should be adhered to when implementing an Ed-Fi REST API over an Ed-Fi application or data store.

Logical Database Model **v1.2**
Graphical guide to Ed-Fi Logical Database Model, based on a relational database schema that is aligned with the reference Ed-Fi Unifying Data Model.

Sandbox

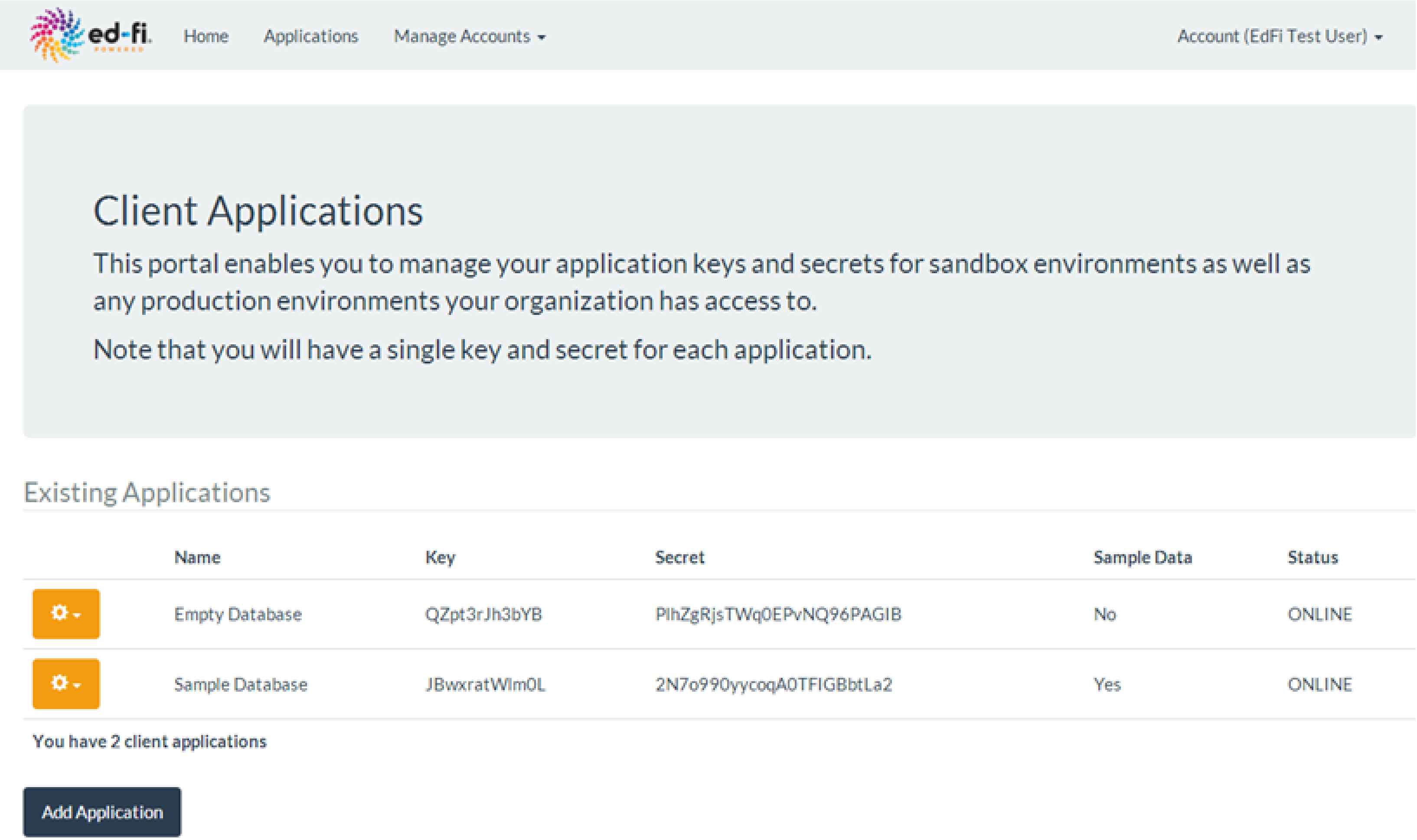
- Sandbox provided to support vendor testing and verification
- Important to test with test data
- Certification will employ a set of automated tests
  - Certification test plan developed with each vendor
- Access to Sandbox
  - Swagger UI: https://azsandboxrestui.cloudapp.net/
  - Request a API Key and Secret
  - Use the Key and Secret in the Swagger UI to make the REST API calls

Overview of the Ed-Fi® REST API Sandbox

Administration Portal

- Invitation-based
- Manage sandbox environments
- Obtain API key and secret for production

View sandbox keys and secrets



Client Applications

This portal enables you to manage your application keys and secrets for sandbox environments as well as any production environments your organization has access to.

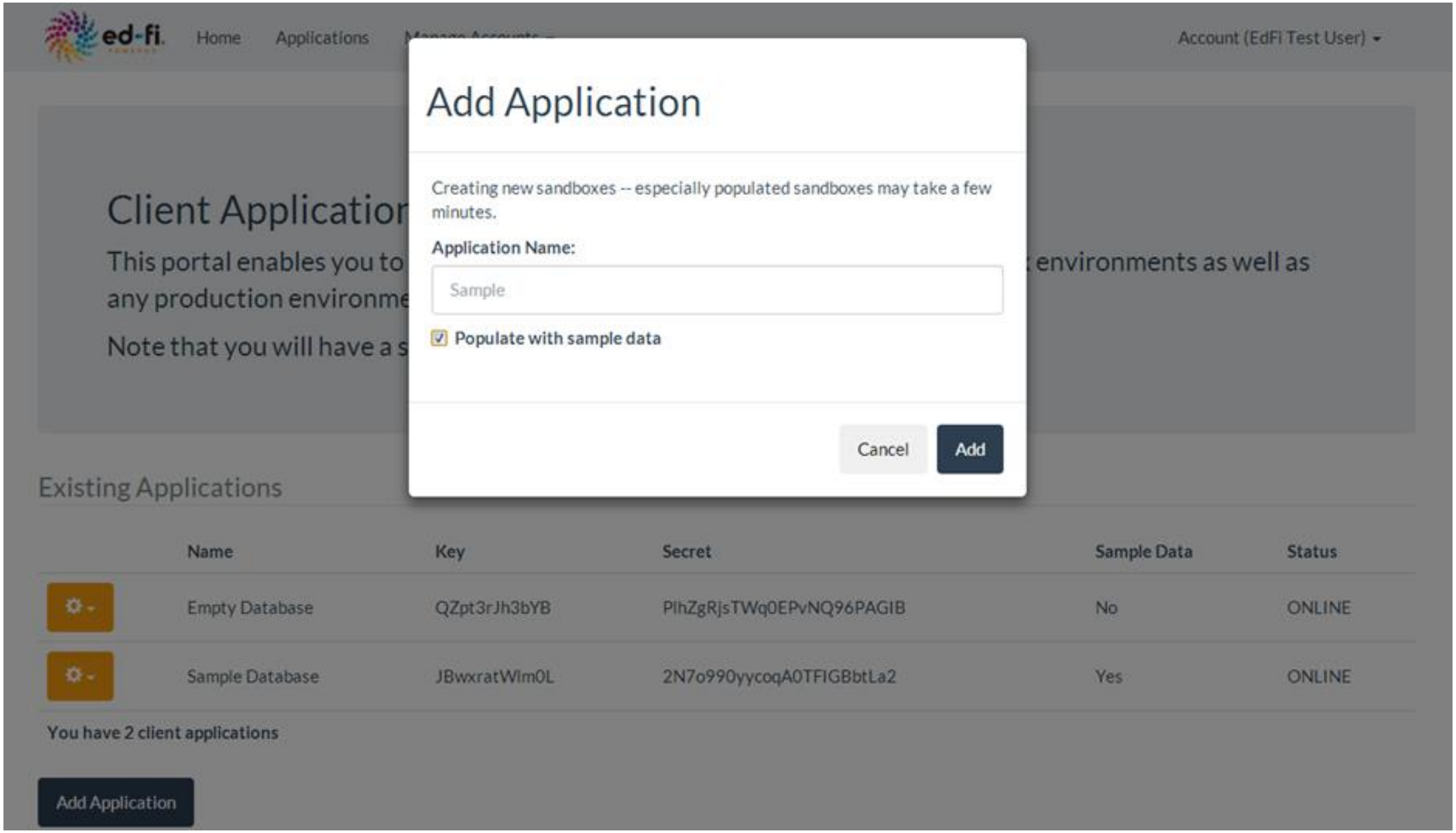
Note that you will have a single key and secret for each application.

Existing Applications

	Name	Key	Secret	Sample Data	Status
⚙ ▾	Empty Database	QZpt3rJh3bYB	PlhZgRjsTWq0EPvNQ96PAGIB	No	ONLINE
⚙ ▾	Sample Database	JBwxratWlm0L	2N7o990yycoqA0TFIGBbtLa2	Yes	ONLINE

You have 2 client applications

Add a new sandbox with or without sample data



Client Application

This portal enables you to … environments as well as any production environme…

Note that you will have a s…

Add Application

Creating new sandboxes -- especially populated sandboxes may take a few minutes.

Application Name: Sample

☑ Populate with sample data

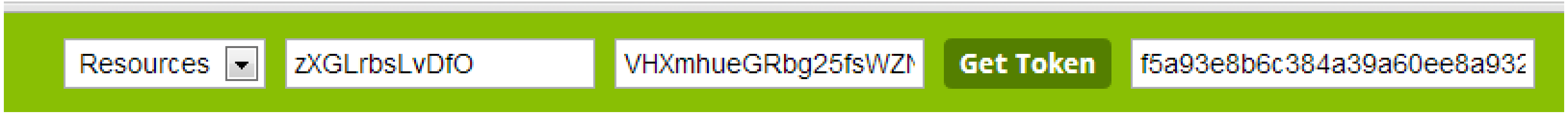
Existing Applications

	Name	Key	Secret	Sample Data	Status
⚙	Empty Database	QZpt3rJh3bYB	PlhZgRjsTWq0EPvNQ96PAGIB	No	ONLINE
⚙	Sample Database	JBwxratWlm0L	2N7o990yycoqA0TFIGBbtLa2	Yes	ONLINE

You have 2 client applications

OAuth2 Authentication

Use the application key and secret from the Administration Portal to obtain an access token



Resources

zXGLrbsLvDfO

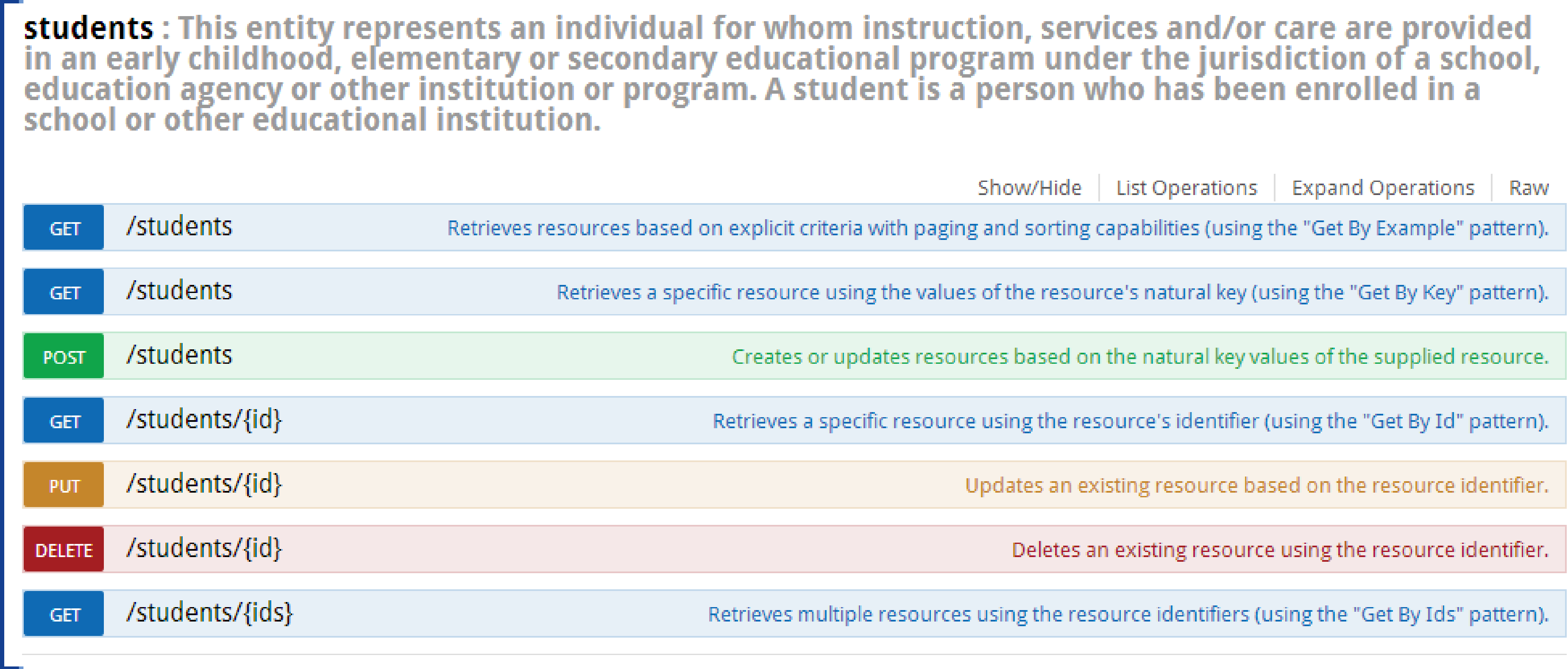
VHXmhueGRbg25fsWZN

f5a93e8b6c384a39a60ee8a932

Online Documentation

- Interactive documentation for the Ed-Fi REST API via Swagger
- Data requirements available through ADE Data Dictionary

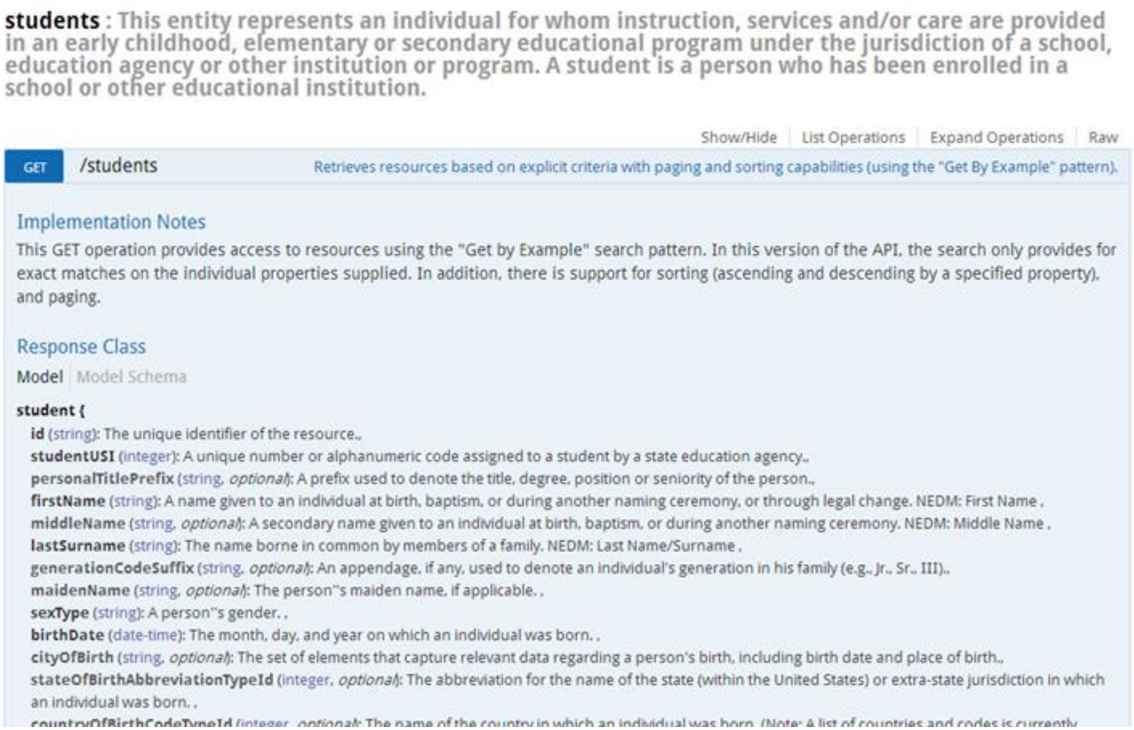
Documentation of each resource



students : This entity represents an individual for whom instruction, services and/or care are provided in an early childhood, elementary or secondary educational program under the jurisdiction of a school, education agency or other institution or program. A student is a person who has been enrolled in a school or other educational institution.

GET	/students	Retrieves resources based on explicit criteria with paging and sorting capabilities (using the "Get By Example" pattern).
GET	/students	Retrieves a specific resource using the values of the resource's natural key (using the "Get By Key" pattern).
POST	/students	Creates or updates resources based on the natural key values of the supplied resource.
GET	/students/{id}	Retrieves a specific resource using the resource's identifier (using the "Get By Id" pattern).
PUT	/students/{id}	Updates an existing resource based on the resource identifier.
DELETE	/students/{id}	Deletes an existing resource using the resource identifier.
GET	/students/{ids}	Retrieves multiple resources using the resource identifiers (using the "Get By Ids" pattern).

Inline documentation



students : This entity represents an individual for whom instruction, services and/or care are provided in an early childhood, elementary or secondary educational program under the jurisdiction of a school, education agency or other institution or program. A student is a person who has been enrolled in a school or other educational institution.

/students — Retrieves resources based on explicit criteria with paging and sorting capabilities (using the "Get By Example" pattern).

Implementation Notes

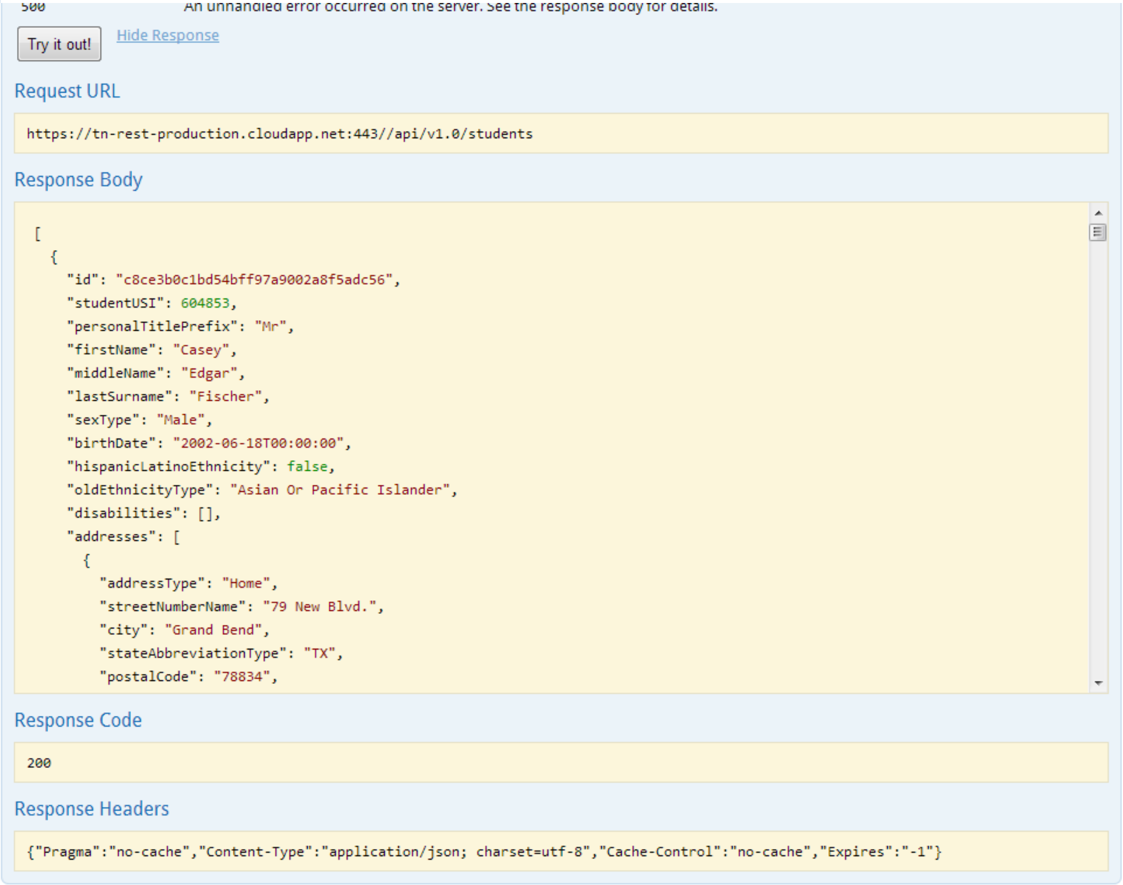
This GET operation provides access to resources using the "Get by Example" search pattern. In this version of the API, the search only provides for exact matches on the individual properties supplied. In addition, there is support for sorting (ascending and descending by a specified property), and paging.

Response Class

```
student {
  id (string): The unique identifier of the resource.,
  studentUSI (integer): A unique number or alphanumeric code assigned to a student by a state education agency.,
  personalTitlePrefix (string, optional): A prefix used to denote the title, degree, position or seniority of the person.,
  firstName (string): A name given to an individual at birth, baptism, or during another naming ceremony, or through legal change. NEDM: First Name ,
  middleName (string, optional): A secondary name given to an individual at birth, baptism, or during another naming ceremony. NEDM: Middle Name ,
  lastSurname (string): The name borne in common by members of a family. NEDM: Last Name/Surname ,
  generationCodeSuffix (string, optional): An appendage, if any, used to denote an individual's generation in his family (e.g., Jr., Sr., III).,
  maidenName (string, optional): The person''s maiden name, if applicable. ,
  sexType (string): A person''s gender. ,
  birthDate (date-time): The month, day, and year on which an individual was born. ,
  cityOfBirth (string, optional): The set of elements that capture relevant data regarding a person's birth, including birth date and place of birth.,
  stateOfBirthAbbreviationTypeId (integer, optional): The abbreviation for the name of the state (within the United States) or extra-state jurisdiction in which
  an individual was born. ,
  countryOfBirthCodeTypeId (integer, optional): The name of the country in which an individual was born. (Note: A list of countries and codes is currently
```

Descriptions provided for each resource and element.

Read sample data



500 An unhandled error occurred on the server. See the response body for details.

Request URL

```
https://tn-rest-production.cloudapp.net:443//api/v1.0/students
```

Response Body

```
[
  {
    "id": "c8ce3b0c1bd54bff97a9002a8f5adc56",
    "studentUSI": 604853,
    "personalTitlePrefix": "Mr",
    "firstName": "Casey",
    "middleName": "Edgar",
    "lastSurname": "Fischer",
    "sexType": "Male",
    "birthDate": "2002-06-18T00:00:00",
    "hispanicLatinoEthnicity": false,
    "oldEthnicityType": "Asian Or Pacific Islander",
    "disabilities": [],
    "addresses": [
      {
        "addressType": "Home",
        "streetNumberName": "79 New Blvd.",
        "city": "Grand Bend",
        "stateAbbreviationType": "TX",
        "postalCode": "78834",
```

Response Code

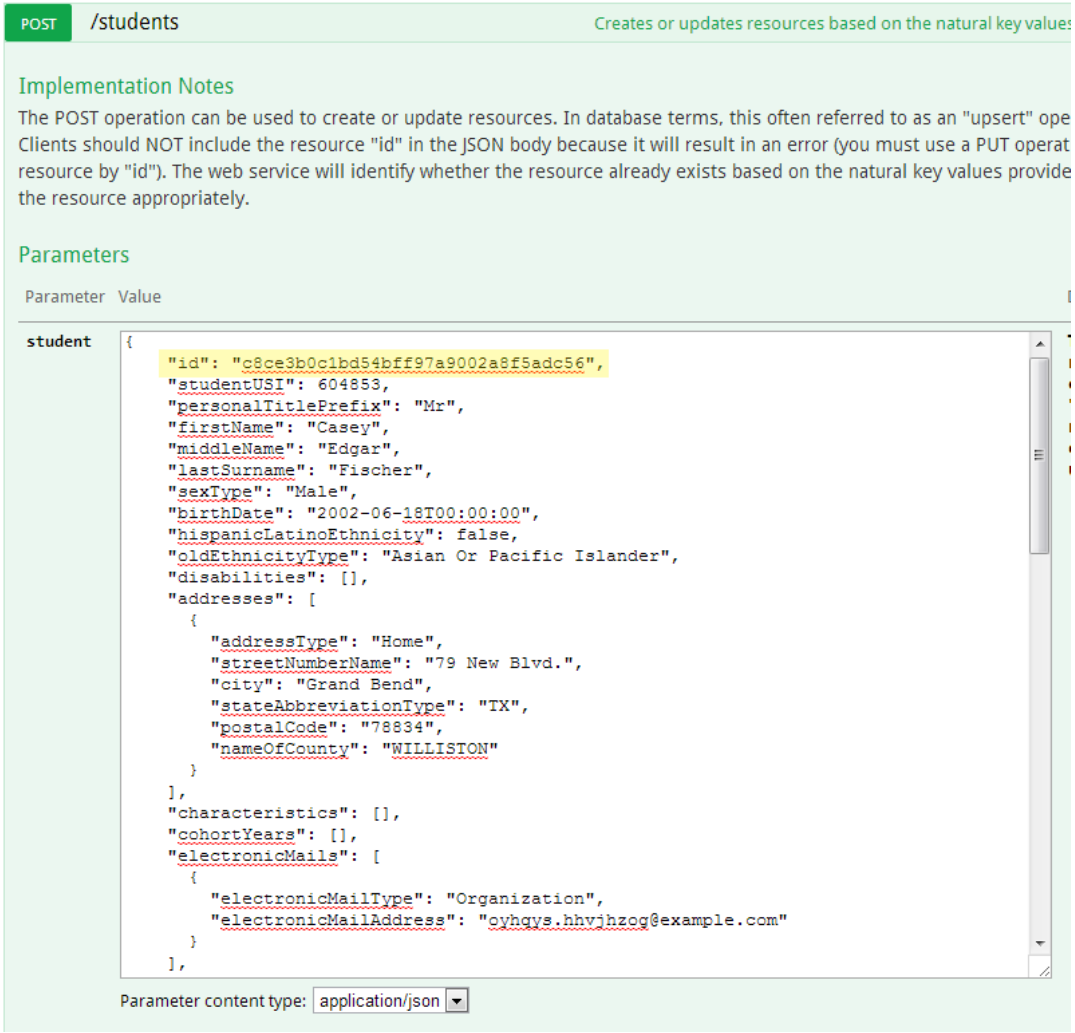
```
200
```

Response Headers

```
{"Pragma":"no-cache","Content-Type":"application/json; charset=utf-8","Cache-Control":"no-cache","Expires":"-1"}
```

Use the GET to obtain the JSON representation of a resource.

Write sample data



POST
/students
Creates or updates resources based on the natural key values

Implementation Notes

The POST operation can be used to create or update resources. In database terms, this often referred to as an "upsert" ope
Clients should NOT include the resource "id" in the JSON body because it will result in an error (you must use a PUT operat
resource by "id"). The web service will identify whether the resource already exists based on the natural key values provide
the resource appropriately.

Parameters

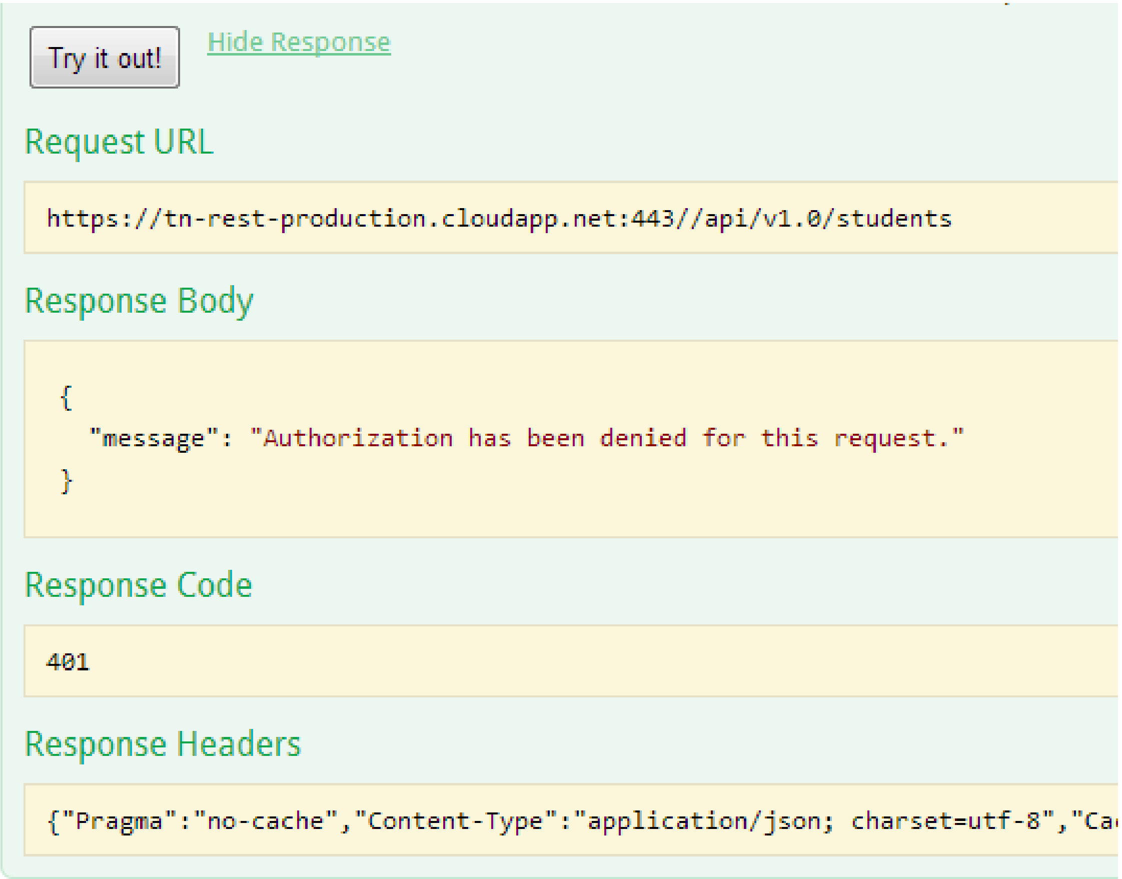
Parameter	Value
student	(JSON below)

```
{
    "id": "c8ce3b0c1bd54bff97a9002a8f5adc56",
    "studentUSI": 604853,
    "personalTitlePrefix": "Mr",
    "firstName": "Casey",
    "middleName": "Edgar",
    "lastSurname": "Fischer",
    "sexType": "Male",
    "birthDate": "2002-06-18T00:00:00",
    "hispanicLatinoEthnicity": false,
    "oldEthnicityType": "Asian Or Pacific Islander",
    "disabilities": [],
    "addresses": [
      {
        "addressType": "Home",
        "streetNumberName": "79 New Blvd.",
        "city": "Grand Bend",
        "stateAbbreviationType": "TX",
        "postalCode": "78834",
        "nameOfCounty": "WILLISTON"
      }
    ],
    "characteristics": [],
    "cohortYears": [],
    "electronicMails": [
      {
        "electronicMailType": "Organization",
        "electronicMailAddress": "oyhqys.hhvjhzog@example.com"
      }
    ],
```

Parameter content type: application/json

Using the data from a GET, copy & paste the JSON into a POST or PUT to write the modified data back into the data store.

Checking for errors



Request URL

```
https://tn-rest-production.cloudapp.net:443//api/v1.0/students
```

Response Body

```
{
  "message": "Authorization has been denied for this request."
}
```

Response Code

```
401
```

Response Headers

```
{"Pragma":"no-cache","Content-Type":"application/json; charset=utf-8","Ca
```

Error messages, if any, are displayed along with the HTTP response code.

Webinars

- Weekly webinars hosted by DLP
  - Vendor support during development and testing of extractors is critical

Contacts @ ADE

Arizona Education Data Standards (AzEDS)

- http://www.azed.gov/aelas/azeds/

By Email at:

- azeds@azed.gov

By Telephone

- (602) 542-5154

Thank You

Arizona
Department of Education